Album Cover Generation from Genre Tags

Alexander Hepburn, Ryan McConville, and Raúl Santos-Rodríguez

University of Bristol,
United Kindgom
{ah13558,ryan.mcconville,enrsr}@bristol.ac.uk

Abstract. This paper presents a method for generating album cover art by including side information regarding the music content. In this preliminary work, using state of the art Generative Adversarial Networks (GAN), album cover arts are generated given a genre tag. In order to have a sufficient dataset containing both the album cover and genre, the Spotify API was used to create a dataset of 50,000 images separated into 5 genres. The main network was pre-trained using the One Million Audio Cover Images for Research (OMACIR) dataset and then trained on the Spotify dataset. This is shown to be successful as the images generated have distinct characteristics for each genre and minimal repeated textures. The network can also distinguish which genre a generated image comes from with an accuracy of 35%.

Keywords: Cover art generation, Generative Adversarial Networks, Genre classification

1 Introduction

Music and visual effects are often linked together and can provide a multi-sensual experience to the user. Given a small set of songs, images could be chosen or created by hand to fit a song or album. However, this becomes unrealistic as the size of the music collection increases. For instance, consider popular website platforms in which users upload their own music. For instance, `SoundCloud`[1] has 12 hours of music and audio uploaded every minute, from over 150 million independent users. Also, a great deal of effort has been recently devoted to the automatic generation of music using deep learning [3]. For example, `Jukedeck`[2] is a platform that uses deep neural networks to generate unique songs from a user specified style and feeling. Stock images are currently displayed but it is more aesthetically pleasing to have a unique image for each song that also reflects some characteristics found within the music. This paper aims to address this by automatically generating an image at the same time as the music content, where the image is unique and reflects some of the characteristics of music. As musical genres are common proxies to categorise and describe music, we use genre labels as a first abstraction of music properties.

[1] `https://soundcloud.com/`
[2] `https://www.jukedeck.com/`

2 Image generation using Generative Adversarial Networks

State-of-the-art approaches in image generation include those based on Generative Adversarial Networks (GANs)[2, 1, 4]. The framework is based on two complementary networks, namely a discriminative (D) network which tries to classify data into sets and a generative (G) network which is used to create new data from a prior distribution. In GANs a generative and discriminative neural network is pitted against each other, posing a minimax problem. New images are generated using G, using noise samples as a seed. Then the newly generated images are used as an input along with real images to D. D then tries to classify which data is real or generated. The variables in D are optimised to be able to distinguish between real and generated data, whilst the variables in G are optimised to fool D into classifying the generated data as real. As such, G learns how to create real looking data simultaneously as D learns to discriminate between generates images and images from the dataset.

In a similar fashion, a Deep Convolutional Generative Adversarial Network (DCGAN) is a GAN which makes use of convolution layers [6]. This can either be in just the generator or both the generator and discriminator. DCGANs achieve better results when generating complex images. Using a combination of up-sampling and transpose convolution layers in the generator produces higher resolution images that look more lifelike.

Finally, the recent Auxiliary Classifier Generative Adversarial Network (AC-GAN)[5] code some descriptive variables into the noise which is used as an input to the generator network. The discriminator then tries to predict these descriptive variables resulting in more consistent training of both the networks as well as being able to specify classes of images. Additionally, they also introduce the use of latent variables in order to make training GANs more consistent. These are random variables that are generated for every generated image and used within the noise vector as input to the generator network. The discriminator then predicts what the random variables used to generate the image are. The use of these latent variables as well as class labels to conditionally generate examples lead to more realistic images as well as being able to generate any class from the pre-specified set of classes.

3 Experiments

Our first goal in this work is to empirically show that it is possible to automatically generate album covers using GANs. As compared to standard image and computer vision datasets, album covers have a huge variety of objects in them as well as different art styles. The limited availability of labelled training data is also a challenge. Finally, we will show how to use AC-GANs to incorporate the genre information into the generation process.

3.1 Genre agnostic generation: One Million Audio Cover dataset

The One Million Audio Cover Images for Research (OMACIR) is a dataset constructed from a variety of sources containing over one million album cover arts[3]. These images are a mixture of greyscale and RGB images, all of different sizes. There are also a large number of repeated images throughout the dataset which would strongly affect any image generation algorithm. A hash based technique was used to detect and remove 798982 duplicate images. All images were resized to 64x64 and standardised so that values lie in the region (-1,1) with a mean of 0. To generate images from an AC-GAN network trained on the OMACIR dataset which lacks classes, we had to modify the cost function to only optimise w.r.t. generating realistic images and predicting latent variables.[4]

Table 1. Network architectures used in the AC-GAN network when generating album covers from both the One Million Cover Images for Research dataset and Spotify dataset, both using 2 latent variables. In the discriminator fully connected 1 is responsible for predicting whether an image is generated or from the dataset, fully connected 2 is responsible for predicting the class label and fully connected 3 is responsible for predicting the latent variables. Transposed convolution is often referred to as deconvolution.

Generator

Layer	Input	Filter	Output	Upsampling	Activation
Fully connected 1	1x100	100x16384	1x16384	0	Linear
Reshape	1x16384		4x4x1024	0	
Transpose Convolution 1	4x4x1024	4x4x512	8x8x512	2	ReLU
Transpose Convolution 2	8x8x512	4x4x256	16x16x256	2	ReLU
Transpose Convolution 3	16x16x256	4x4x128	32x32x128	2	ReLU
Transpose Convolution 4	32x32x128	4x4x3	64x64x3	2	Tanh

Discriminator

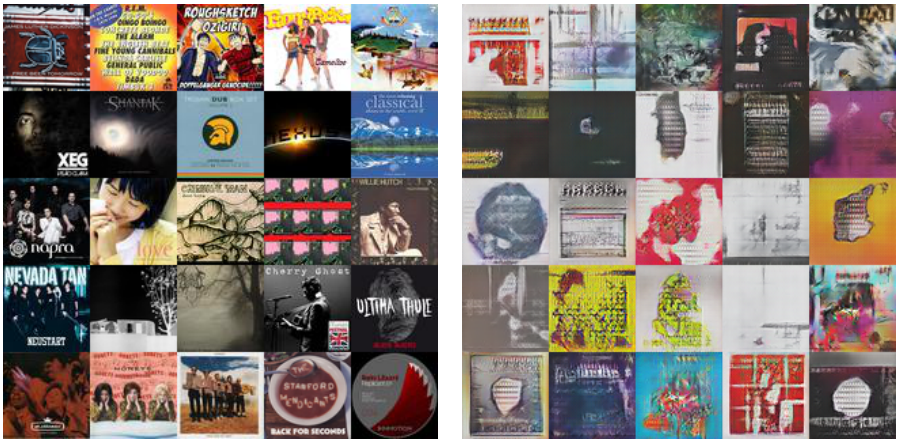
Layer	Input	Filter	Output	Stride	Activation
Convolution 1	64x64x3	4x4x128	32x32x128	2	Leaky ReLU
Convolution 2	32x32x128	4x4x256	16x16x256	2	Leaky ReLU
Convolution 2	16x16x256	4x4x512	8x8x512	2	Leaky ReLU
Convolution 2	8x8x512	4x4x1024	4x4x1024	2	Leaky ReLU
Reshape	4x4x1024		1x16384	0	
Fully connected 1	1x16384	16384x1	1x1	0	Linear
Fully connected 2	1x16384	16384x5	1x5	0	Linear
Fully connected 3	1x16384	16384x2	1x2	0	Linear

The network architecture used is detailed in Table 1. The best network parameters, found via a grid search, include a generative learning rate of 0.002, a discriminative learning rate of 0.001 and a batch size of 128. The input noise

[3] `https://archive.org/details/audio-covers`

[4] Code can be found at `https://github.com/alexhepburn/cover-art-generation`.

is taken from a uniform distribution in the region (-1, 1). Overall the resulting images in Fig. 1 are of good visual quality with minimal repeated textures and have properties which are indicative of album covers.



(a) Original album covers. (b) Generated album covers.

Fig. 1. AC-GAN trained on the OMACIR dataset.

3.2 Genre aware generation: Spotify dataset

Although OMACIR is extremely useful due to the amount of images, it contains no metadata of artists, genres or album names. To compile a dataset that contains such metadata, the Spotify API[5] was queried with a variety of genres (Jazz, Dance, Rock, Rap and Metal) and the first 10,000 unique album names were recorded for each genre. While it has been established that a deep learning network can generate realistic looking album cover art from the OMACIR dataset, our objective is to generate album covers given prior knowledge about the album itself. To do so requires the use of an AC-GAN network whereby the genre is the descriptive variable used. In order to decrease overfitting an AC-GAN network was first pre-trained using OMACIR and then trained using the Spotify dataset. A discriminative learning rate of $2 \cdot 10^{-5}$, a generative learning rate of $1 \cdot 10^{-5}$ and a batch size of 128 were found to be optimal.

One major flaw when training AC-GANs is that the generator may collapse and always output the same image. One popular method of tracking diversity amongst classes is the use of multi-scale structural similarity (MS-SSIM) [7]. MS-SSIM is an extension of the well known structural similarity index. A high MS-SSIM index for a generated class indicates that there is little variation amongst generated images and as such the generator has collapsed. The MS-SSIM scores

[5] `https://developer.spotify.com/web-api/`

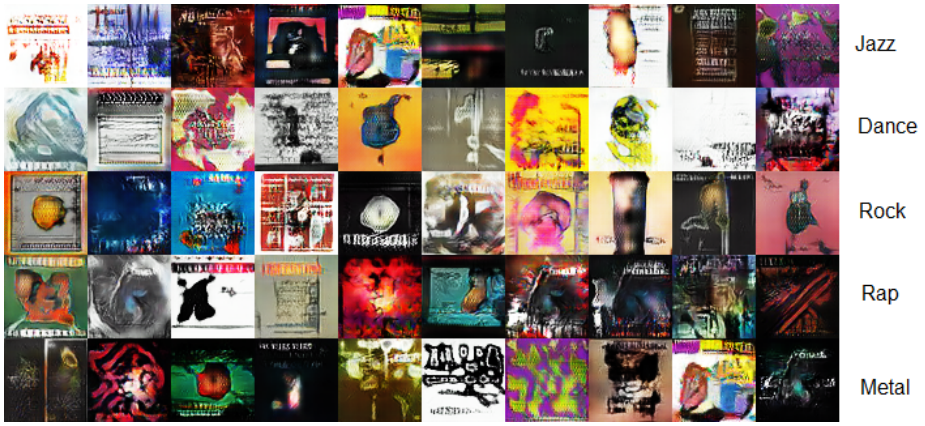
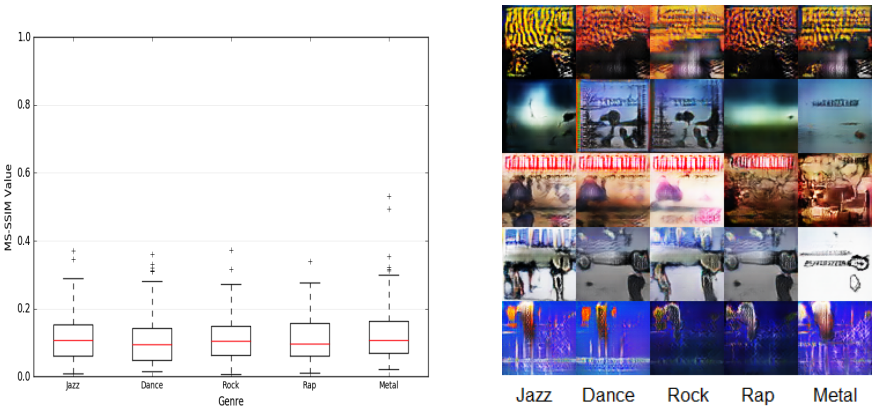


Fig. 2. Images generated from the final AC-GAN network. The network was pre-trained using the OMACIR dataset and then trained using the Spotify dataset.

between real and generated images within the same genre have a similar distribution to the MS-SSIM scores between just the real images, as shown in Fig. 3(a). This means that in terms of MS-SSIM, the real and generated images are interchangeable without affecting the MS-SSIM distribution. Although variance within classes is important, perhaps more important is being able to distinguish which class an image is generated from. Given a generated image, the cross-validated discriminator accuracy for genre classification is $35 \pm 2\%$. For images from the Spotify dataset, the network is able to correctly predict the genre with an accuracy of $47 \pm 4\%$. To establish a baseline for predicting genres of an album cover, a separate network was trained to predict which genre a real album cover belonged to. The network has the same architecture as the discriminator detailed in Table (1) and has a cross-validated accuracy of $59 \pm 4\%$. This implies that there can be improvements in combining both classifying genres and generating images into one network. To explore the visual characteristics of each class, images were generated using the same random and latent variables but with different genres. Fig. 3(b) shows that changing the genre has a different effect depending on the image, although general trends can be spotted. For example, rap covers are noticeably darker while jazz albums are overall lighter. Jazz and rap have more soft light colours whereas the rest have more black harsh shapes, however they all have a similar colour palette. This means the image structure or colour palette is represented in the latent and random variables whereas the style is specified by the genre. This is a positive result as different genres can use the same objects on their album covers but they each have an distinguishable style to them.



(a) MS-SSIM for each genre between 1000 real and 1000 generated examples.

(b) Effect of using same noise and latent variables but different genre.

Fig. 3. Genre diversity of images generated from the AC-GAN network.

4 Conclusions

We have explored the conditional generation of album cover art using AC-GAN architectures, using genre labels in the process. Overall the conditional generation of 64x64 album covers given a genre is possible, although there are still repeated textures in the new images. Using AC-GANs opens up opportunities to include more information about albums when generating cover art although larger images will need to be generated for this to become feasible for a platform such as SoundCloud.

References

1. Denton, E.L., Chintala, S., Fergus, R., et al.: Deep generative image models using a laplacian pyramid of adversarial networks. In: NIPS (2015)
2. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS (2014)
3. Huang, A., Wu, R.: Deep learning for music. arXiv:1606.04930 (2016)
4. Im, D.J., Kim, C.D., Jiang, H., Memisevic, R.: Generating images with recurrent adversarial networks. arXiv:1602.05110 (2016)
5. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier gans. arXiv:1610.09585 (2016)
6. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: ICLR (2016)
7. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: Asilomar Conference on Signals, Systems and Computers. vol. 2, pp. 1398–1402 (2003)